

## PDF RENDERIZATION OF Example2.R OUTPUT

```
# Example 2: logistic models with multilevel categorical covariates
# Real data from Spain's Encuesta Financiera de Familias (2002-2020)

library(data.table)
library(OaxacaSurvey)
library(fastDummies)

## Thank you for using fastDummies!

## To acknowledge our work, please cite the package:

## Kaplan, J. & Schlegel, B. (2023). fastDummies: Fast Creation of Dummy (Binary) Columns and Rows from

# Import dataset with multilevel categorical data from a s
df <- fread("tests/eff-pool-2002-2020.csv")
df <- df[sv_year == 2020]
df[, group := 0][class == "worker", group := 0][class == "capitalist", group := 1]
df[, rentsbi := 0][rents >= renthog * 0.1 & rents > 2000, rentsbi := 1]
df[homeowner == "", homeowner := "Non-Owner"]
df$class <- relevel(as.factor(df$class), ref = "self-employed")
df$bage <- relevel(as.factor(df$bage), ref = "0-34")
df$inherit <- relevel(as.factor(df$inherit), ref = "Non-inherit")
df$homeowner <- relevel(as.factor(df$homeowner), ref = "Non-Owner")
df$riquezafin <- factor(as.logical(df$riquezafin), levels = c(TRUE, FALSE), labels = c("Fin", "NonFin"))
total_variables <- c(
  "facine3", "renthog", "renthog1", "bage", "homeowner", "worker", "young", "sex", "class",
  "actreales", "riquezanet", "riquezafin", "educ", "auton", "class",
  "tipo_auton", "direc", "multipr", "useprop", "inherit"
)
selected_variables <- c(
  "renthog", "bage", "sex", "homeowner", "riquezafin"
)

#####

# select regressors
data <- df[, ..selected_variables]

# transform multi-level categories to dummies. IMPORTANT remove both first dummy and selected column to
data <- fastDummies::dummy_cols(data,
  select_columns = c("bage", "sex", "homeowner", "riquezafin"),
  remove_first_dummy = TRUE,
  remove_selected_columns = TRUE
)

# name the columns as x1...xn for oaxaca-blinder survey
colnames(data) <- paste0("x", seq_along(colnames(data)))
length_reg <- length(colnames(data))
new_formula <- paste("y ~", paste0("x", 1:length_reg, collapse = " + "))

# compose the final dataset by adding: engenuous variable, groups of split and sample weights
data <- cbind(y = df$rentsbi, group = df$group, weights = df$facine3, data)
```

```

# finally, test the model
result2 <- oaxaca_blinder_svy(
  as.formula(new_formula),
  data = data.frame(data),
  group = "group",
  weights = "weights",
  R = 1000
)

```

```

## Warning in coef(model1)[-1] - coef(model2)[-1]: longer object length is not a
## multiple of shorter object length

```

```

## Warning in coef(model1)[-1] - coef(model2)[-1]: longer object length is not a
## multiple of shorter object length

```

```

## Warning in coef(model1)[-1] - coef(model2)[-1]: longer object length is not a
## multiple of shorter object length

```

```

## Warning in coef(model1)[-1] - coef(model2)[-1]: longer object length is not a
## multiple of shorter object length

```

```

## Warning in coef(model1)[-1] - coef(model2)[-1]: longer object length is not a
## multiple of shorter object length

```

```

## Warning in coef(model1)[-1] - coef(model2)[-1]: longer object length is not a
## multiple of shorter object length

```

```

## Warning in coef(model1)[-1] - coef(model2)[-1]: longer object length is not a
## multiple of shorter object length

```

```

## Warning in coef(model1)[-1] - coef(model2)[-1]: longer object length is not a
## multiple of shorter object length

```

```

## Warning in coef(model1)[-1] - coef(model2)[-1]: longer object length is not a
## multiple of shorter object length

```

```

## Warning in coef(model1)[-1] - coef(model2)[-1]: longer object length is not a
## multiple of shorter object length

```

```

## Warning in coef(model1)[-1] - coef(model2)[-1]: longer object length is not a
## multiple of shorter object length

```

```

## Warning in coef(model1)[-1] - coef(model2)[-1]: longer object length is not a
## multiple of shorter object length

```

```

## Warning in coef(model1)[-1] - coef(model2)[-1]: longer object length is not a
## multiple of shorter object length

```

```

## Warning in coef(model1)[-1] - coef(model2)[-1]: longer object length is not a
## multiple of shorter object length

```

```

## Warning in coef(model1)[-1] - coef(model2)[-1]: longer object length is not a
## multiple of shorter object length

```











```
## Warning in coef(model1)[-1] - coef(model2)[-1]: longer object length is not a
## multiple of shorter object length

## Warning in coef(model1)[-1] - coef(model2)[-1]: longer object length is not a
## multiple of shorter object length

## Warning in coef(model1)[-1] - coef(model2)[-1]: longer object length is not a
## multiple of shorter object length

## Warning in coef(model1)[-1] - coef(model2)[-1]: longer object length is not a
## multiple of shorter object length

## Warning in coef(model1)[-1] - coef(model2)[-1]: longer object length is not a
## multiple of shorter object length

## Warning in coef(model1)[-1] - coef(model2)[-1]: longer object length is not a
## multiple of shorter object length

## Warning in coef(model1)[-1] - coef(model2)[-1]: longer object length is not a
## multiple of shorter object length

## Warning in coef(model1)[-1] - coef(model2)[-1]: longer object length is not a
## multiple of shorter object length
```

```
print(result2)
```

```
##           unex           end           coef           inter           total  means1_y
## value -0.01650112 -0.006674515  0.178472290 -0.12452758  0.03076908 0.09638539
## CI1   -0.15236347 -0.016691176 -0.008801649 -0.22666504 -0.01777712 0.04955993
## CI2    0.12357768  0.004117406  0.350821203  0.03434048  0.10554340 0.15552057
##           means2_y  means_dif
## value 0.06854613  0.02783927
## CI1   0.05979796 -0.01777712
## CI2   0.07840142  0.08664588
```